# 3D Quick Start Guide:

## 3ds max™ to Director 8.5 Shockwave Studio®

discreet™

macromedia

**Trademarks**

Afterburner, AppletAce, Attain, Attain Enterprise Learning System, Attain Essentials, Attain Objects for Dreamweaver, Authorware, Authorware Attain, Authorware Interactive Studio, Authorware Star, Authorware Synergy, Backstage, Backstage Designer, Backstage Desktop Studio, Backstage Enterprise Studio, Backstage Internet Studio, Design in Motion, Director, Director Multimedia Studio, Doc Around the Clock, Dreamweaver, Dreamweaver Attain, Drumbeat, Drumbeat 2000, Extreme 3D, Fireworks, Flash, Fontographer, FreeHand, FreeHand Graphics Studio, Generator, Generator Developer's Studio, Generator Dynamic Graphics Server, Knowledge Objects, Knowledge Stream, Knowledge Track, Lingo, Live Effects, Macromedia, Macromedia M Logo & Design, Macromedia Flash, Macromedia Xres, Macromind, Macromind Action, MAGIC, Mediamaker, Object Authoring, Power Applets, Priority Access, Roundtrip HTML, Scriptlets, SoundEdit, ShockRave, Shockmachine, Shockwave, Shockwave Remote, Shockwave Internet Studio, Showcase, Tools to Power Your Ideas, Universal Media, Virtuoso, Web Design 101, Whirlwind and Xtra are trademarks of Macromedia, Inc. and may be registered in the United States or in other jurisdictions including internationally. Other product names, logos, designs, titles, words or phrases mentioned within this publication may be trademarks, servicemarks, or tradenames of Macromedia, Inc. or other entities and may be registered in certain jurisdictions including internationally.

This guide contains links to third-party Web sites that are not under the control of Macromedia, and Macromedia is not responsible for the content on any linked site. If you access a third-party Web site mentioned in this guide, then you do so at your own risk. Macromedia provides these links only as a convenience, and the inclusion of the link does not imply that Macromedia endorses or accepts any responsibility for the content on those third-party sites.

Discreet is a division of Autodesk, Inc.  Discreet, 3ds max, Biped, character studio, and Physique are either registered trademarks or trademarks of Discreet Logic Inc./Autodesk, Inc. in the U.S.A. and/or other countries.

Discreet Disclaimer

DISCREET, A DIVISION OF AUTODESK, INC., MAKES NO WARRANTIES, EITHER EXPRESS OR IMPLIED, REGARDING THE ENCLOSED COMPUTER SOFTWARE PACKAGE, ITS MERCHANTABILITY OR ITS FITNESS FOR ANY PARTICULAR PURPOSE.  THE ABOVE EXCLUSION MAY NOT APPLY TO YOU.  THIS WARRANTY PROVIDES YOU WITH SPECIFIC LEGAL RIGHTS.  THERE MAY BE OTHER RIGHTS THAT YOU MAY HAVE WHICH VARY FROM STATE TO STATE.

**Apple Disclaimer**

**APPLE COMPUTER, INC. MAKES NO WARRANTIES, EITHER EXPRESS OR IMPLIED, REGARDING THE ENCLOSED COMPUTER SOFTWARE PACKAGE, ITS MERCHANTABILITY OR ITS FITNESS FOR ANY PARTICULAR PURPOSE. THE EXCLUSION OF IMPLIED WARRANTIES IS NOT PERMITTED BY SOME STATES. THE ABOVE EXCLUSION MAY NOT APPLY TO YOU. THIS WARRANTY PROVIDES YOU WITH SPECIFIC LEGAL RIGHTS. THERE MAY BE OTHER RIGHTS THAT YOU MAY HAVE WHICH VARY FROM STATE TO STATE.**

**Copyright © 2001 Macromedia, Inc. All rights reserved. This manual may not be copied, photocopied, reproduced, translated, or converted to any electronic or machine-readable form in whole or in part without prior written approval of Macromedia, Inc.**

Macromedia, Inc.
600 Townsend St.
San Francisco, CA 94103

# CONTENTS


Macromedia Director Quick Start Guide . . . . . 5

Introduction . . . . . .5
Tutorial Part I: Creating 3D Assets . . . . . .6
Animating the character, ball and camera . . . . . .10
Tutorial Part II: The **3ds max** Exporter to Director 8.5
Shockwave Studio . . . . . .13
Tutorial Part III: Assembling the Assets in Director 8.5
Shockwave Studio . . . . . .17
Tutorial Part IV: How to publish to the Web mainstream with the
Shockwave 3D Player. . . . . .26

# Macromedia Director Quick Start Guide

## Introduction

This 3D Quick Start Guide is designed for the multimedia designer who is new to the 3D capabilities of Director® Shockwave® Studio and the 3D artist/animator who is unfamiliar with Director® 8.5 Shockwave® Studio. Our goal today is to animate an American-style football player, kicking the ball over the goal post. This character was created in **3ds max™ 4** and animated using **character studio® 3**, proven software solutions from Discreet. You will find a number of assets in the .zip file. By the end of this tutorial, you will be able to take these assets into Director 8.5 Shockwave Studio and create an interactive 3D sequence, playable on the web. The player is kicking the ball over the goal post. The character is animated to perform a series of motions in sequence along a single timeline, i.e., an idle motion, a jog start, a kick sequence, a sprint with arms up and a jog stop. The football is animated to leave its start position, fly over the goal and bounce on the other side. The goal is to take these assets into Director Shockwave Studio and to make them interactive.

*Figure 1*

The tutorial begins with 3D creation tips and tricks for preparing the character mesh, skinning it onto the Biped® skeleton using Physique® and animating the character. Both Biped and Physique are **character studio** features. It then discusses the **3ds max** Shockwave Exporter to Director 8.5 Shockwave Studio in the "Director 8.5 Shockwave Studio Environment" options and what happens when you export 3D content. Next, in the Director Shockwave Studio environment you will import and assemble the elements (i.e., the ground, goal post, environment, ball, light, camera and character). Finally, you will add interactivity using Director Shockwave Studio Behaviors and Lingo™.

During the tutorial, you will

1. Import the .w3d file and set up the scene
2. Use the Property Inspector and Director Shockwave Studio Behaviors to regulate the lights and camera
3. Use Lingo to trigger the run and kick animation
4. Use Lingo to make the arrow keys run the character around the playing field

# Tutorial Part I: Creating 3D Assets

## Modeling the character using Multiresolution Mesh and Hierarchical Subdivision Surfaces technology

The new 3D engine in Director 8.5 Shockwave Studio includes Multiresolution Mesh and Hierarchical Subdivision Surfaces Technology. Multiresolution Mesh provides continuous polygon decimation, allowing a model's detail to be reduced proportionally on the fly. Subdivision Surfaces provides selective polygon tessellation, permitting a model's detail to be increased on the fly. Previously, real-time 3D applications required using low-polygon models to optimize game engine performance running with limited resources. The typical scene polygon "budget" previously was 3000 polygons, a main character budget was 1200-1500 and a minor character budget was under 500. Also, artists typically made multiple instances of each character at different resolutions (known as levels of detail). Different levels of detail were swapped in/out of the scene based on the distance from the camera, allowing for a more optimum usage of scene budget.

**3ds max 4** software contains both the Multiresolution Mesh and Subdivision Surfaces modifiers. These advances permit characters to be modeled with significantly higher budgets, because the polygon count is reduced via Multiresolution Mesh or increased via Subdivision Surfaces in Director Shockwave Studio. The character in this exercise was modeled at high resolution and textured in pieces in **3ds max**. It was then assembled as a series of meshes before the Physique was applied. You can make some adjustments when you make your model in **3ds max**. Below are screen shots which show the effect of the Multiresolution Mesh and Subdivision Surfaces modifiers.

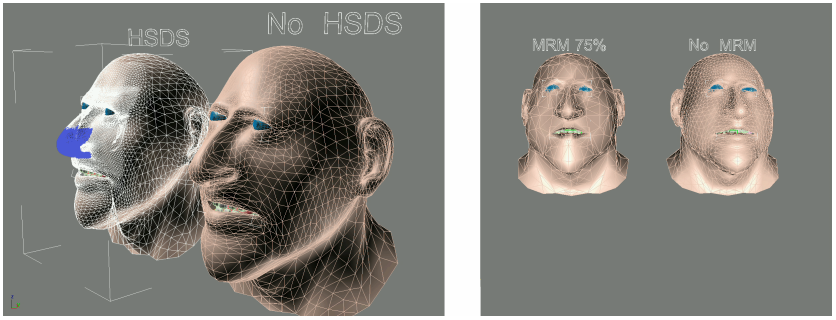



*Figure 2, figure 3*

Here Multiresolution Mesh reduced the face count by 75 percent, with comparatively little impact on the rendered image above.

Subdivision Surfaces subdivides polygons where a selection is made or as needed to add detail and smooth curves. Note how Subdivision Surfaces successively subdivides the mesh as you move from back to front.

## Polygonal models are preferred

Director Shockwave Studio's 3D Engine supports polygons that are viewable in the Shockwave export window. In **3ds max**, this means both the Editable Mesh and Editable Polygons states, and any other polygonal forms collapse to Editable Mesh or Editable Polygon. Although Patches and NURBS are not supported, you can often model with these geometries, and then collapse the results to an editable mesh. Pay close attention to seams when using these geometries, as they may become visible or split when viewed either in a **3ds max** render of the model or in the Shockwave window.

## Animated modifiers will not export; Animated Hierarchies Require Groups

Modifiers are available to deform or model a mesh, but an animated modifier will not export. This applies to simple modifiers such as Bend, Stretch, Displace or FFD, as well as Morpher. All animated mesh deformations in this release must be bone-driven. This means you must bone a bird's wings to animate its bending; likewise, you must use bones on the character's head to perform facial animation. **character studio**'s Biped is the only supported bone system.

3D-enhanced Shockwave Player collapses hierarchies that are not bone-based. So, for example, if you want to animate a character's eyes moving back and forth while its head moves up and down, you must use groups. Group each eye and the head separately, then parent each eye group to the head group and animate the groups. Grouping prevents the collapse and the sub-level animations survive.

## Texturing the model: some map channels are not supported

In **3ds max**, only the Blinn shader is supported. On export, this is converted to Gouraud. Only the Standard and Multi/Sub-Object materials are supported; there is no support for multi-layered or animated maps. Most Material settings will export properly. Values in the Blinn Basic Parameters rollout, (e.g., Ambient, Diffuse, Specular, Self-Illumination, Opacity, Specular Level, Glossiness) export properly. There is limited Map Support; you can use any Bitmap or 2D procedural map. In this tutorial, the textures on the player's body are Multi/Sub-Object materials. Note that the map tree is only one level deep and restricted to certain channels.

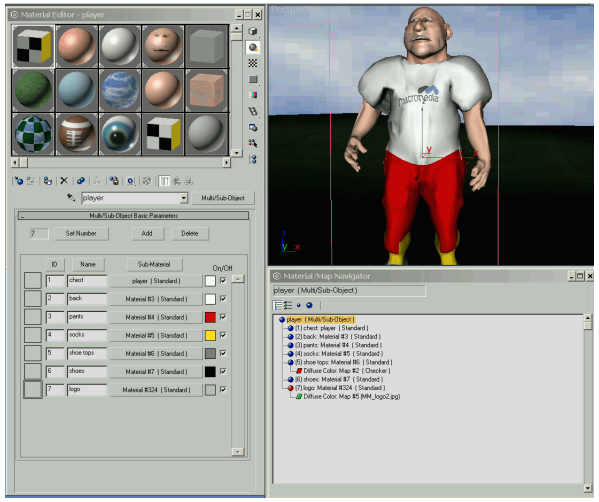



*Figure 4*

There is limited Map Channel Support; the above maps can be used in the Diffuse and Reflection Map Channels. Bump Maps will export in the .w3d file, but are not supported in this version of the Shockwave 3D Player. Opacity maps can be used only in the form of an alpha channel included with the bitmap in the Diffuse Map Channel. If you want the texture to include transparency, you must use a Targa, TIFF, or other bitmap containing an alpha channel. Maps in other channels may export, but the results will be different than they appear in **3ds max**.

If you have textures which will not export (i.e., unsupported channels, multi-layered textures, animated textures, etc.) consider applying them in Director 8.5 Shockwave Studio.

## UVW coordinates apply

Shockwave 3D Player supports UVW texture coordinates implemented through the UVW Map modifier, but not through **3ds max** software's Material Editor. Fine tune positioning, scaling and tiling using the modifier only. If you need to "Decal" a texture map, use a Multi-Sub-Object material or, alternatively, decal the map in Director.

Since UVW coordinates are supported, specialized modifiers or third-party paint and mapping programs can get professional results. This includes **3ds max**'s Unwrap UVW and Textporter modifiers, as well as high-end commercial plug-ins, such as Deep Paint™ 3D with Texture Weapons. The character's face was touched up in Deep Paint 3D.

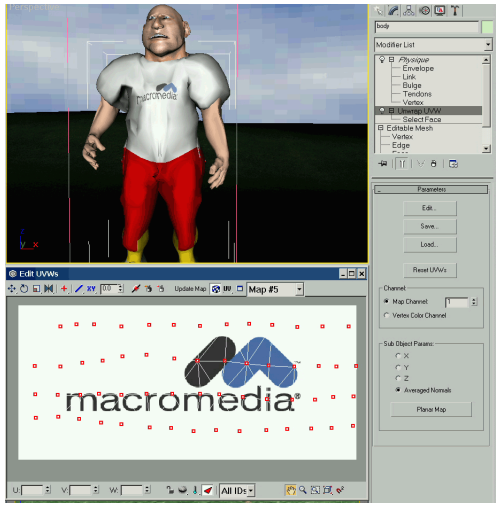
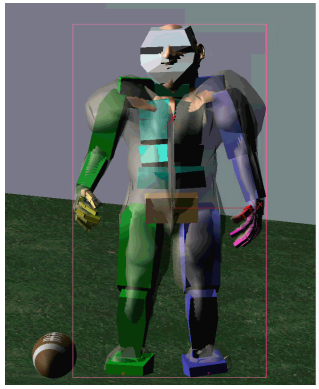



*Figure 5*

# Animating the character, ball and camera

## Bones and Biped animation

To animate a mesh deformation, the Shockwave Player requires a Biped character, which uses a bone-based hierarchy. Discreet's **character studio** is necessary to implement bones animation in Shockwave 3D technology. Even if you do not use the Biped, you will need the Physique modifier to skin the character. **3ds max**'s Skin modifier is not supported. The Physique modifier binds the mesh to the Biped bones (see Figure 7 below). You will need to fine-tune the Physique modifier by adjusting the Envelope Sub-Objects. First do this in the stationary "Reference Pose" and then check it against the animation by leaving the Reference Mode and scrubbing the animation bar.



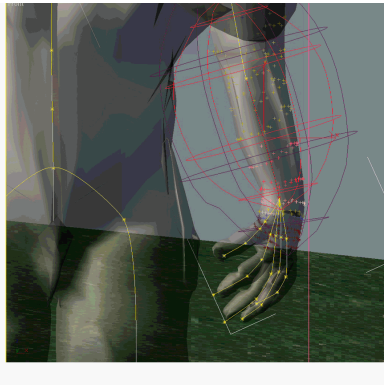


*Figure 6, figure 7*

You will need patience and practice to master Biped and Physique. They are worth using, however, because they permit recycling of motions among characters, animation layering, as well as importing and editing motion capture data usable in Director Shockwave Studio (8.5). Physique envelopes radiate from each bone link and compete in their influence for each skin vertex, so skin slides across bones as in real life. Since Director 8.5 Shockwave Studio is compatible with **character studio 3** characters, you can use **character studio 3** and get faithful recreation of the model and motion in Director Shockwave Studio (8.5). On export, **3ds max** Exporter to Director Shockwave Studio will convert the mechanics **3ds max** uses to animate the character (i.e. bones, inverse kinematics, etc.) into a keyframed hierarchy that is recognized by Director Shockwave Studio.

## The motion timeline

You may want several animation sequences for an animated real-time character. You have three methods available to accomplish this: (1) make copies of the character in the same file, applying different animations to each; (2) make separate files, each with a copy of the character performing a different motion, and export only the animation; or (3) have the character perform a series of motions along a single timeline, so you can "pull out" and loop specific motions by designating a range of frames. The enclosed **3ds max** motion file contains several motions strung together—an idle pose, a jog to run, a run and kick, a run with hands up in a victory sign and a jog stop. By paying attention to the frame where each sequence starts and stops, you can use Director Shockwave Studio to interactively sample, loop and direct the running of the animation.

## Animating the ball

The ball flight and bounce is a keyframed animation. You can trigger this sequence to run normally in Director Shockwave Studio, or use Lingo to trigger it to begin when the character gets within a certain distance of the ball.

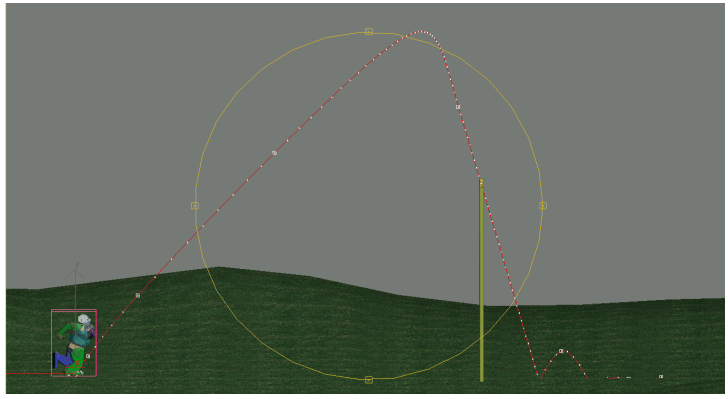


*Figure 8*

Alternatively, you could use the Havok™ physics to implement gravity, kick-collision, and ground collision to show how the ball bounces. This would be a simulation.

## Animating the camera

Camera animations in **3ds max** do not currently export to Director Shockwave Studio (8.5). Instead, once inside Director Shockwave Studio, you can use Director Shockwave Studio Behaviors to animate the camera.

## User-defined settings in 3ds max

You can set streaming priorities by modifying an object's "User Defined Properties", accessed by a right-click on a mesh and selecting "Properties/User Defined". The Lingo commands you can enter are set forth in detail in the **3ds max** Exporter User Guide, in the Translators folder on the Director 8.5 Shockwave Studio installation CD. The properties window is simply a notepad for entering the code to which the Exporter will respond. If you were concerned with the order in which items stream into this scene, we would set that priority here.

## More 3D resources

The Macromedia site provides several resources for your 3D education and on how to implement 3D in Director Shockwave Studio. You might start with "Preparing 3D content for Shockwave 3D Player", found at www.macromedia.com/support/director/work_3d/models_use_in_sw/models_use_in_sw.html. This article gives an overview of the issues 3D artists can expect in preparing 3D content for Shockwave 3D Player and cross-references other sources for those who want to dig deeper. Follow this with "Getting oriented to 3D", found at www.macromedia.com/support/director/3d_lingo/getting_oriented_with_3d/getting_oriented_with_3d.html. This article presents some of the basic concepts you will need to understand to make use of 3D cast members in your movies. Each of these concepts is also illustrated in sample movies.

# Tutorial Part II: The 3ds max Exporter to Director 8.5 Shockwave Studio

Once you have finished the scene in **3ds max**, you can export in whole or in part to Director Shockwave Studio using the **3ds max** Exporter to Director Shockwave Studio. By default it sends the entire scene to the .w3d file and provides feedback regarding what was written to it. You have the option to limit the export, both in terms of which assets are included and the compression assigned to those assets. The exporter functions are covered in greater depth in the **3ds max** Exporter User Guide (on the Director 8.5 installation CD)

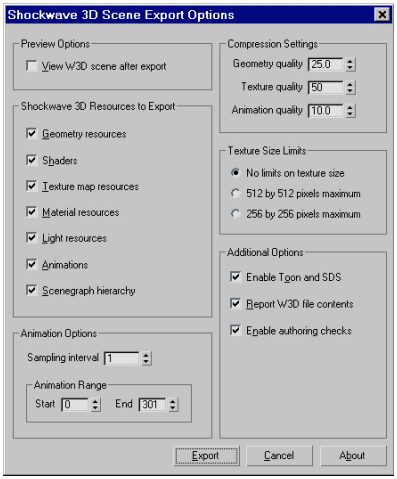



*Figure 9*

## Turn on View W3D

Checking the View W3D checkbox gives a post-export view of what was written to the export file. It shows the exported objects, textured, animated and rendered using the Shockwave 3D renderer. You can navigate the window with zoom, pan and orbit functions, and will see animations playing as they will within Director Shockwave Studio. This may look different from what you see in your **3ds max** viewport or rendered test movies, because the Shockwave 3D renderer uses a Gouraud Shader. The Exporter screen grab, below, shows how your scene will appear in Director Shockwave Studio.

The View window is navigable in **3ds max** as follows:

```
Left Mouse (LM) = Orbit/Rotate
LM + Space Bar = Pan
LM + Alt = Roll Up/Down
LM + Shift = Roll Right/Left
LM + Ctrl = Dolly/Zoom
```

*Figure 10*

When you close the Export View window, you find the Shockwave 3D File Analysis graph, displaying the total file size; its composition in geometry, textures, and animation; and the initial file load size (ILS), which is important for streaming purposes.

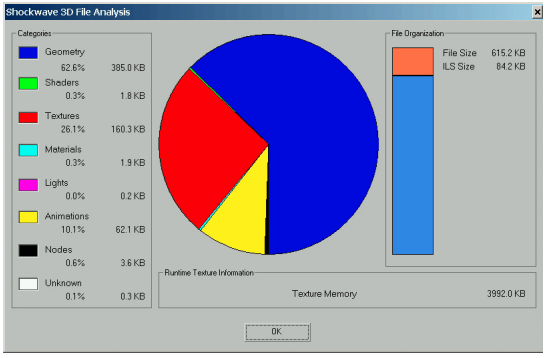



*Figure 11*

The final window you see on export contains warning and error messages relating to what was just written to the .w3d file. Note that some comments are merely warnings, which may make no difference to file performance depending on the platform.

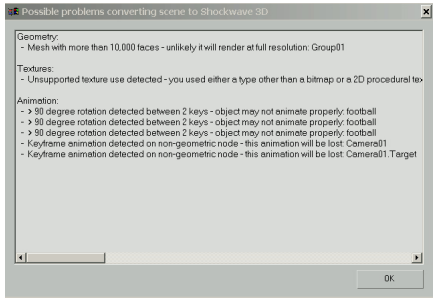

Possible problems converting scene to Shockwave 3D
Geometry:
- Mesh with more than 10,000 faces - unlikely it will render at full resolution: Group01
Textures:
- Unsupported texture use detected - you used either a type other than a bitmap or a 2D procedural tex
Animation:
- > 90 degree rotation detected between 2 keys - object may not animate properly: football
- > 90 degree rotation detected between 2 keys - object may not animate properly: football
- > 90 degree rotation detected between 2 keys - object may not animate properly: football
- Keyframe animation detected on non-geometric node - this animation will be lost: Camera01
- Keyframe animation detected on non-geometric node - this animation will be lost: Camera01.Target
OK

*Figure 12*

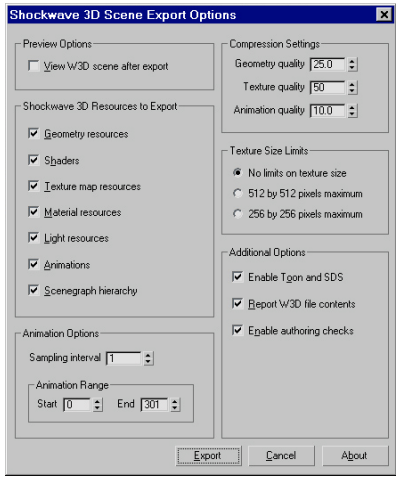



*Figure 13*

## Choose the Shockwave 3D Resources to Export

Typically you will export the entire scene. But you can limit the file size. For example, if you were sending out additional motions for use in an existing world, you would check only "Animations". If you were building a material library in Director Shockwave Studio, you would send out "Shaders", "Textures and Map Resources" and "Material Resources".

## How Much of the Animation to Export?

You can limit the range and sampling interval with the Shockwave 3D Exporter. In this case, and most of the time, you will export the entire range at an interval of 1. If the animation is jerky, you can use a smaller sample. If the .w3d file is too large, you can increase the sample interval.

## How Much Compression?

One of the advantages of Shockwave 3D technology is the ability to deliver high quality media in a small bundle. The exported .w3d file contains polygonal geometry, textures (converted to JPEG) and animation data. You can add sound, physics and interactivity to the package in Director Shockwave Studio.

The scale 0–100 in this section is not linear and requires testing to achieve the optimum balance between file size and detail. In this tutorial, the geometry is dense; setting compression at 50 (down from 100) increases Multiresolution Mesh compression and thereby the agility of the character in the virtual world. On the other hand, the animation component is complex. A loss of detail here significantly compromises movement. The texture maps are very simple and are substantially compressed without noticeable change (i.e., 25 down from 100).

These settings dramatically affect the size and performance of the .w3d file. Texture maps make up a significant portion of the file size and are compressed with JPEG compression. (Note: the Exporter converts all maps to JPEG. A JPEG map compresses more than a Targa or TIFF file. Use a TIFF or Targa bitmap only if your texture requires an alpha channel to vary opacity). Geometry is compressed using Multiresolution Mesh technology.

## Limit Texture Size?

The dimensions of the texture map can affect performance as much as compression. Shockwave 3D can handle maps over 1024x1024, but you should find 256x256 sufficient for almost all tasks. This exporter setting will convert large maps to low-resolution on export.

## Which Additional Options?

The additional options allow you to drop Subdivision Surfaces support in Director Shockwave Studio and to turn off the additional reporting dialogs. Turn these off only if you need to minimize the .w3d file size and you have optimized all other settings.

# Tutorial Part III: Assembling the Assets in Director 8.5 Shockwave Studio

You cannot send a .w3d file straight to the web. You must import it into Director Shockwave Studio and save it out as a Director movie (DIR, DCR) before you can display a Shockwave 3D file in your browser. The file comes in as an Internal Cast Member.

Double click on this file to open the 3D Shockwave Viewer. You can play the animation. You can navigate the scene. (If you get lost you can reset the camera or the entire 3D world.)

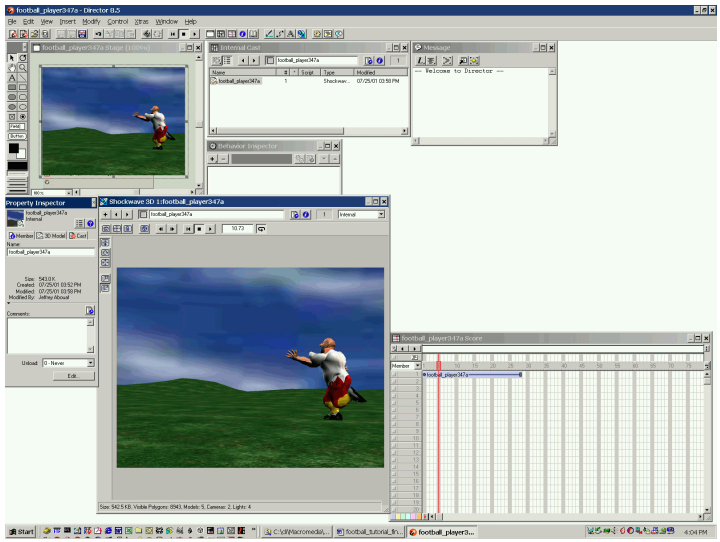
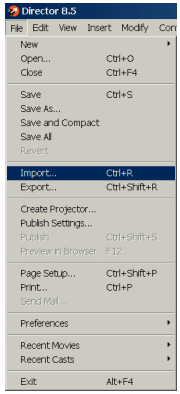


*Figure 14*



## Positioning and orienting the playing field, character and ball

The Shockwave 3D Viewer and Property Inspector let you visually explore the world and its contents. To create a movie, drag the animation (now a 3D internal cast member) to the stage. If you save the movie at this point, you can load and play the movie in any browser enabled with Shockwave Player.

*Figure 15*

## Setting up lights and camera

You can set up lights using the viewer window and the Property Inspector. Here you can vary the position, color and intensity of the illumination. No Behaviors are needed for these modifications. You can control camera orbit using the drag and drop Behaviors. Click the Library icon (the book icon on the main toolbar) to call up Library Actions. (See Figure 16.)

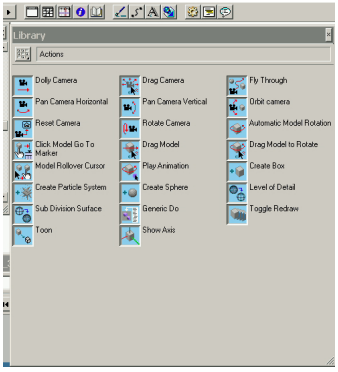



*Figure 16*

1 Under Actions, select Orbit camera and drag it onto the Stage.

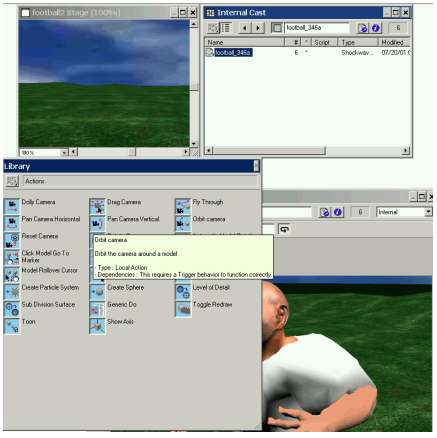



*Figure 17*

2 Next choose the Mouse Right button trigger from the Triggers palette.

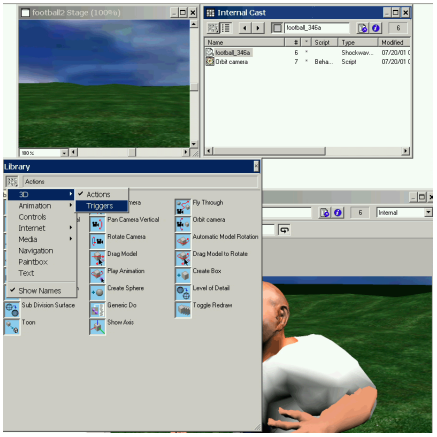



*Figure 18*

3 Choose "Anytime…" and "Group Unassigned – Camera orbit on X and Y"

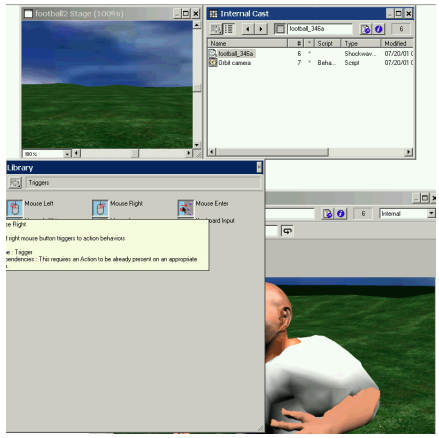



*Figure 19*

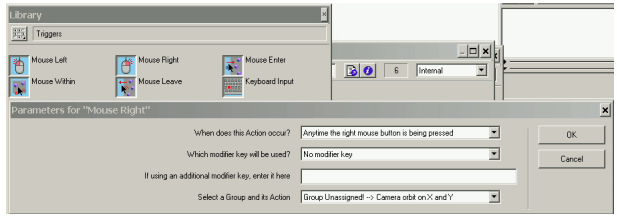



*Figure 20*

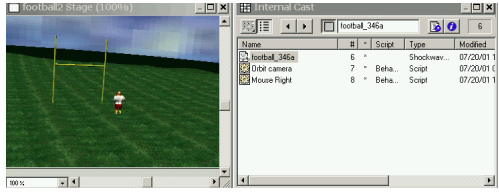



*Figure 21*

## Triggering the run and kick sequences

The animation sequence begins with an idle position, followed by run, jog and kick motions, then a run, arm raise, and finally, a jog-stop. The ball animation is in sync with the kick animation. We use simple Lingo command to trigger the animation through the ball kick.

To start, set the cast member's looping playback option to FALSE. This is done using the Property Inspector. Select the cast member, open the Property Inspector, uncheck the "loop" option, or set it to FALSE.

In the beginSprite handler, for the player model, queue up the "Group01" motion file to loop over a short sequence. This will be the "run" animation while under user control later. Pause both models' animations by setting their modifier playRate values to 0.0. Now the models are paused and waiting to go.

Initialize a property, pAnimState. It's used to track what "state" of playback you are in. It starts at a value of #paused to indicate that you are in the initial paused state. When the user clicks with the mouse, if the state is #paused, then the animations start by setting the playRates of the models to 1.0. Then update the property's value to #playback (indicating fixed playback of the run and kick motion sequence). At the end of the animations, a handler is triggered once again to pause the animation and update the pAnimState property to #userControl. This indicates that the user can now manipulate the character.

The motions are paused by setting a flag value to indicate the pause. The first mouse click triggers the motion playback and resets the flag value. When the animation ends, this property value is reset again to indicate that the user is under control.

## Making the character run using the arrow keys

You can use Lingo to sample and loop the run portion from the animation sequence to create a sustained running motion. Then use additional Lingo scripting to enable the user to vary the run speed and direction, and prevent the character from running off the field.

Open the file "football_interactive.dir." You can view the script using the Script Inspector. This script uses a keyDown handler that is triggered every time the user presses a key (or is triggered multiple times if the user presses and holds the key). Every time a key is pressed, it checks to see which key it was and then responds accordingly. For example, the left/right arrows rotate the model left or right. The up/down arrows increase or decrease the modifier's playRate property, resulting in a slowdown or acceleration in the motion. The following sample code is commented to provide you with additional information. Review it and play with alterations to vary the interaction.

The following Lingo controls both:

- Triggering the animation with the left-mouse and after the first animation is done.
- Running the character using the arrow keys.

Although these events logically follow one another in sequence, the Lingo code that controls this is not organized in the same linear manner.

```
property pMember -- reference to the member used by this sprite
property pSprite -- reference to this sprite

property pBall -- reference to the football model
property pPlayer -- reference to the kicker model

property pAnimState -- symbol indicating the current animation
state
property pCoordinateBoundaries -- property list containing the
coordinate
               -- boundaries within which to constrain the
               -- character model's movement under user control

property pInitZ

on beginSprite (me)

   -- initialize basic properties
   pMember = sprite(me.spriteNum).member
   pSprite = sprite(me.spriteNum)

   -- reset the member's world
   pMember.resetWorld()

   -- initialize node related properties
   pBall = pMember.model("football")
   pPlayer = pMember.model("Group01")

   -- initialize each model
   pBall.keyFramePlayer.positionReset = FALSE
   pBall.keyFramePlayer.rotationReset = #none
   pBall.keyFramePlayer.queue("football-Key",0)
   pBall.keyFramePlayer.pause()

   pPlayer.bonesPlayer.positionReset = FALSE
   pPlayer.bonesPlayer.queue("Group01",0)
   pPlayer.bonesPlayer.queue("Group01",1,7730,8730,1.0,7730)
   pPlayer.bonesPlayer.pause()

   -- register the model to receive the animation ended event
   pPlayer.registerScript(#animationEnded,#animEnded,me)

   -- initialize animation state property
   pAnimState = #paused
   pCoordinateBoundaries = [:]
   pCoordinateBoundaries.addProp(#x,[#max: 3.7e3, #min: -3.7e3])
   pCoordinateBoundaries.addProp(#y,[#max: 1.5e3, #min: -5.6e3])

   tPosn = pPlayer.worldPosition
```

```
    pInitZ = tPosn.z

end beginSprite

on exitFrame (me)

    -- grab the character's current world position
    currentPosition = pPlayer.worldPosition
    newPosition = currentPosition.duplicate()

    -- check player's x position for being within the boundaries
    currentX = currentPosition.x
    if (currentX > pCoordinateBoundaries.x.max) then
        newPosition.x = pCoordinateBoundaries.x.max
    end if
    if (currentX < pCoordinateBoundaries.x.min) then
        newPosition.x = pCoordinateBoundaries.x.min
    end if

    -- check player's y position for being within the boundaries
    currentY = currentPosition.y
    if (currentY > pCoordinateBoundaries.y.max) then
        newPosition.y = pCoordinateBoundaries.y.max
    end if
    if (currentY < pCoordinateBoundaries.y.min) then
        newPosition.y = pCoordinateBoundaries.y.min
    end if

    newPosition.z = pInitZ

    -- apply new position if necessary
    if (currentPosition <> newPosition) then
        pPlayer.worldPosition = newPosition
    end if
end exitFrame

on mouseUp (me)

    -- check to see if the animations are paused
    if (pAnimState = #paused) then

        -- begin animation playback for both models
        pBall.keyFramePlayer.playNext()
        pBall.keyFramePlayer.play()
        pPlayer.bonesPlayer.playNext()
        pPlayer.bonesPlayer.play()

        -- update the animation state
        pAnimState = #playback

    end if
```

```
end mouseUp

on animEnded (me,a,b,c,d)

   -- pause the player's motion via the playRate property
   pPlayer.bonesPlayer.playRate = 0.0

   -- update the animation state property
   pAnimState = #userControl

end animEnded

on keyDown (me)

   -- only respond if the character is under user control
   if (pAnimState = #userControl) then

      -- check the keyCode
      case (the keyCode) of

         123:  -- left arrow

            -- rotate the character

pPlayer.rotate(pPlayer.worldPosition,vector(0,0,1),5,#world)

         124:  -- right arrow

            -- rotate the character
            pPlayer.rotate(pPlayer.worldPosition,vector(0,0,1),-
5,#world)

         125:  -- down arrow

            -- get the current playRate
            currPlayRate = pPlayer.bonesPlayer.playRate

            -- decrease the player's playRate if needed
            if (currPlayRate > 0.0) then
               newPlayRate = currPlayRate - 0.2
               if (newPlayRate < 0) then newPlayRate = 0
               pPlayer.bonesPlayer.playRate = newPlayRate
            end if

         126:  -- up arrow

            -- get the current playRate
            currPlayRate = pPlayer.bonesPlayer.playRate

            -- increase the player's playRate if needed
            if (currPlayRate < 2.0) then
               newPlayRate = currPlayRate + 0.2
```

```
                if (newPlayRate > 2.0) then newPlayRate = 2.0
                pPlayer.bonesPlayer.playRate = newPlayRate
            end if
        end case

    end if

end keyDown
```

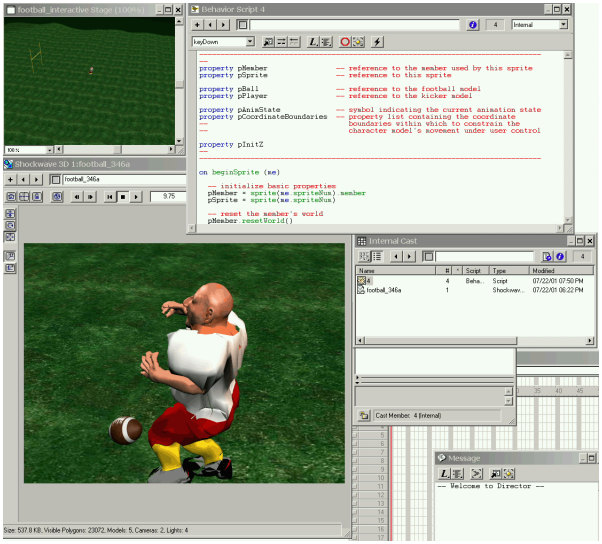


*Figure 22*

## Tutorial Part IV: How to publish to the Web mainstream with the Shockwave 3D Player

After you have brought the cast member on stage, applied Behaviors and any Lingo, you can save your movie out in the .dir format. This can be either a Save As or Save and Compact command. You can then play the movie in any browser enabled with the Shockwave 3D Player. From the browser's Page Source feature, you will find the entire Director Shockwave Studio file is embedded in the browser. Alternatively, you can save your movie using the File > Publish command. This creates the same movie in DCR format. The browser's Page Source will show this simply as a DCR file, and will not expose the underlying Director Shockwave Studio file. Note that the Behavior and Lingo interactivity go with the movie to the browser.